

Europäisches Patentamt
European Patent Office
Office européen des brevets



(11) **EP 0 823 798 A1**

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
11.02.1998 Bulletin 1998/07

(21) Application number: 97401903.6

(22) Date of filing: 08.08.1997

(51) Int Cl.⁶: **H04H 1/00**, H04L 12/18,
H04N 7/173, H04N 7/52,
H04N 7/58
// H04N7/20

(84) Designated Contracting States:
**AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC
NL PT SE**

(30) Priority: 09.08.1996 FR 9610067

(71) Applicant: **THOMSON Multimedia**
92648 Boulogne Cédex (FR)

(72) Inventors:
• **Deniau, Eric**
92648 Boulogne Cédex (FR)
• **Straub, Gilles**
92648 Boulogne Cédex (FR)

(74) Representative: **Ruellan-Lemonnier, Brigitte et al**
THOMSON multimedia,
46 quai A. Le Gallo
92648 Boulogne Cédex (FR)

(54) **Receiver in a cyclic packet data transmission system**

(57) The invention relates to a receiver in a cyclic packet data transmission system, the said system furthermore including at least one transmitter.

According to the invention, the receiver includes:

- means (5) for demultiplexing and filtering the said data packets,
- means (6) for storing a database from data selected from data structures of the packets extracted by filtering,
- means (19, 23) for detecting updating of data structures including data appearing in the database,
- means (19, 23) for comparing, in case of detection of updating of a data structure, the data stored in the database and the corresponding data of the said updated data structure and, only in the case in which there is a difference, for notifying a client application of this difference.

The invention applies particularly within the framework of the transmission of electronic programme guides in DVB ("Digital Video Broadcast") or DSS ("Digital Satellite System") type digital television systems.

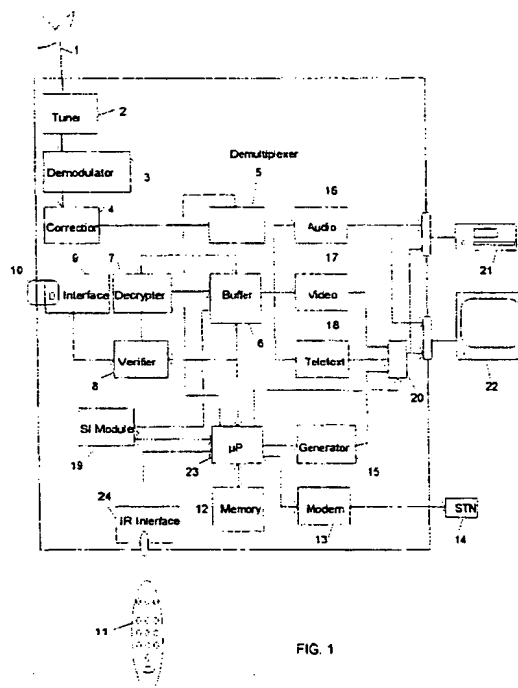


FIG. 1

EP 0 823 798 A1

- Figure 1 is a block diagram of a television receiver implementing the present example embodiment,
- Figures 2a to 2c are timecharts of the exchanges which have taken place between an application, the data management module and the data source, in accordance with the present example,
- Figure 3a, respectively 3b is a state chart illustrating the operation of a one-off, respectively permanent request,
- 5 - Figure 4 is a diagram of a screen of an application, namely an electronic programme guide according to the present example embodiment,
- Figure 5 is a chart of the database maintained by the management module.

It will be observed that for fuller information regarding the format and content of the service data, MPEG and DVB tables and sections, reference will be made in particular to the following three documents:

- (a) ETS 300 468 - Specification for Service Information (SI) in Digital Video Broadcast (DVB) systems - January 23, 1996,
- (b) ISO/IEC 13818-1 (1994) Generic Coding of Moving Pictures and Associated Audio - Recommendation H.220,
- 15 also called "MPEG II Systems", and
- (c) ETR 211 - Digital Broadcasting systems for television: Implementation guidelines for the use of MPEG-2 systems; Guidelines on implementation and usage of service information.

Figure 1 is a block diagram of a DVB (Digital Video Broadcasting) type digital television integrated decoder/receiver. It is obvious that the invention is not limited to this physical environment, but may easily be adapted to another type of service data transmission.

The decoder of Figure 1 is linked to an antenna 1, itself linked to a tuner 2 of the decoder. The signal provided by the tuner is demodulated by a demodulator 3. The demodulated data are corrected by a corrector circuit 4 and transmitted to a demultiplexer 5.

The latter is, for example, a demultiplexer similar to that described in French patent application 95 15767 filed on 29 December 1995 in the name of THOMSON multimedia. The demultiplexer 5 includes a certain number of filters programmed by a microprocessor 23 as a function of the various applications supported by the decoder. For the clarity of the diagram, only the most important connections of the microprocessor 23 are illustrated.

The audio or video packets or sections filtered by the demultiplexer are stored in predefined areas in a buffer memory 6 for the attention of these applications. If necessary, the information is firstly decrypted by a decrypter circuit 7 depending on the user's entitlements, before being stored in this buffer memory 6.

According to the present example, the applications are five in number: an audio decoder 16, a video decoder 17, a Teletext decoder 18, an access control assembly (comprising the decrypter 7, a verifier microcontroller 8 and an interface for a microprocessor card 9 linked in normal operating mode to a microprocessor card 10), as well as a service data management module.

The decoder also includes an infrared interface for a remote control 24, the said interface being likewise linked to the microprocessor 23. The latter is connected to a memory 12 which includes the operating system as well as resident or downloaded programmes for running the applications.

A modem 13 linked to the switched telephone network 14 is also controlled by the microprocessor.

A character generator 15 allows the generation of command or graphics menus relating to the parameters of the decoder or to a particular application. The video signal generated by this character generator is multiplexed with one of the video signals coming from the video decoder 17 or from the Teletext decoder 18 towards a first TV peripheral socket (SCART socket) linked to a television 22 or a second TV peripheral socket linked to a video recorder 21. The multiplexing circuit 20 is managed by the microprocessor 23.

The invention relates more particularly to the operation of the service data management module. In the present case, this module is physically speaking a programme managed by the microprocessor, although conceptually it concerns an application which processes data packets, in the same manner as an audio or video decoder, and for which dedicated circuits are used.

The module is an interface between the service data (MPEG and DVB tables and sections) and client applications (programme guide, telepurchasing, interactive games, etc.). It manages the requests from client applications and maintains an internal database on the strength of the service data received.

According to the present example embodiment, the client application is a programme guide also managed by the microprocessor.

The management module makes a certain number of functions available to the client applications, these functions being intended to formulate the requests relating to the information needed by the applications.

The request functions operate asynchronously. The response to a request, if response there be, is notified to an application by the management module when this response is available. This requires the implementation of a request function identification mechanism. For this purpose, an identifier is chosen by the application for each request issued

includes a list of events which are selected using the said commands. For each of the events, the programme guide displays the title, the name of the corresponding service and the start and finish time.

The upper part 40 can display only a part of the list of events. To access the other events the user uses scrolling arrows of the remote control.

When the programme guide application addresses the requests relating to the information for the events of the list to the management module, the request concerning the events displayed first will be of the non-advance type, that is to say will have priority. This is in fact information which absolutely must be displayed. The request relating to the other events of the list will be of the advance type, and processed by the management module after the non-advance requests. It is not in fact definite that the application will need the corresponding information, since it is not certain that the user will actually scroll the events if the particulars which he is looking for are among the events displayed initially. For the purpose of speeding up the display of these data in the case in which they are requested, they are however preloaded. When the data in response to an advance request have been demultiplexed, the management module notifies the application which launched this request of this fact. The transfer of data to a buffer made available by the application does not however take place so long as the application does not request this transfer.

One of the roles of the service data management module is to programme the filters of the demultiplexer. To fulfil this function and allow fast access to the sought-after data, it maintains in accordance with the present example embodiment an image of the physical structure of the network or networks to which it has access.

Documents a and b define ten tables giving information about the configuration of the network or networks, bundles, services and events transmitted. The tables are identified by particular values of PID (Packet Identification Data) and of table identifiers (table_id), the values of which are defined by the said documents. Each table contains a version identifier making it possible to determine whether from one transmission of the table to another, the content of this table has changed.

A version identifier can also be used at the level of a descriptor or group of descriptors, and can be so used in parallel with the descriptor of the table.

The table which interests us here is the so-called NIT table (standing for Network Information Table). The NIT table includes information about a given transmission network, in particular the list of services available per transmission channel (Transport Stream).

The data management module constructs an internal indexing of the networks, channels and services available. When the decoder is switched on or when the NIT table is updated, a logic key is allocated to each of the services available. This key is the index of this service in the database maintained by the module.

In a DVB system, a service can be located uniquely by the route comprising the following variables:

- network_id (identifier of the network),
- (transport_stream_id; original_network_id) pair
- service_id (identifier of the service proper).

The three variables are natural integers coded on 16 bits.

Three types of lists are created: one list for the networks, one list of channels for each network and one list of services for each channel.

An element in the list of networks is created each time a NIT table which includes a new network is demultiplexed. To do this, the transport packets whose PID is equal to 0x0010 are filtered. These packets actually contain the NIT tables, additionally identified by a variable table_id. A 4-bit code is associated with each network, in the order of the demultiplexing of the corresponding tables. The code is the index of the address pointer of the structure which includes the information relating to this network.

The NIT table includes the list of channels for this network, as well as the list of services available for each channel.

For each network of the list of networks, a list of channels is created. Each element of a list of channels is indexed with the aid of 5 bits. The list contains the address pointers of the structures which include the data specific to each channel. The logic key for identifying a channel in the database is composed of the 4 index bits of the network, followed by the 5 bits of the index of the channel of this network.

For each channel, a list of services is created, containing the identifiers of the services described in the NIT table. Each service in a list is indexed on 7 bits. The logic key of a service in the database therefore includes 16 bits in all: 4 network index bits, 5 channel bits and 7 service bits.

An event of a service will be identified with the aid of the 16 bits denoting this event (variable event_id of the table), to which will be appended the 16 bits of the logic key of the associated service.

The structure of the database (other than events) is organized according to the following structures:

| Channel | |
|--|--|
| ChannelIdentifier ("TransportStream id") | |
| OriginalNetworkIdentifier ("OriginalNetwork id") | |
| ServicesListAddress | |

| ServicesList | |
|--------------|--------------------------|
| 0 | ServiceAddress |
| 1 | ServiceAddress |
| 2 | ServiceAddress |
| 3 | ServiceAddress |
| 4 | ServiceAddress |
| 5 | ServiceAddress |
| 6 | ServiceAddress |
| 7 | ServiceAddress |
| | ServicesArrayNextAddress |

| Service | |
|----------------------------------|--|
| ServiceIdentifier ("service id") | |
| ServiceName ("service name") | |
| Status ("running status")... | |

The variables whose name contains the term "Address" are pointers to memory areas corresponding to the start of a data structure.

The other variables correspond to information extracted from the data stream. To facilitate understanding, these variables are followed in brackets and between quotation marks by the name used in document (a).

It will be noted that the lists of networks, of channels and of services are each organized as arrays, each array being composed on the one hand of eight pointers to data structures of the network, channel or service type, and on the other hand of a pointer to a possible array containing the rest of the list. The latter pointer is null when there is no other array, i.e. when an array contains the last elements of a list.

The Database array contains a pointer to the array containing the first part of the list of networks.

The NetworksList array includes the pointers to the first eight networks. According to the present example embodiment, there are at most two NetworksList arrays, containing the complete list of networks.

The Network array includes the information relating to a given network, as well as a pointer to the list of channels associated with this network.

The structure of the other arrays is similar to what has just been described. It is moreover easy to extend it to the events and to other types of data.

According to a variant embodiment, the requests relating to the data concerning the structure of the network, of the channels and of the services are requests of permanent type, this for the purpose of keeping constantly up to date the image of the network in the database.

In their exchanges with the management module, the applications use only the logic keys. These are translated by the module into a memory address corresponding to the site at which the information is stored.

networks (especially the list of networks and the associated lists of channels) and maintains them in a permanent manner.

It should be noted that the invention is not limited merely to the transmission of data by satellite, radio or cable, but can be employed in any system in which data or data packets appear periodically in the data stream. This is the case in particular for data streams which are recorded and read back.

Claims

1. Receiver in a cyclic packet data transmission system, the said system furthermore including at least one transmitter, characterized in that the said receiver includes:
 - means (5) for demultiplexing and filtering the said data packets,
 - means (6) for storing a database from data selected from data structures of the packets extracted by filtering,
 - means (19, 23) for detecting an update of data structures including data appearing in the database,
 - means (19, 23) for comparing, in case of detection of an update of a data structure, the data stored in the database and the corresponding data of the said updated data structure and, only in the case in which there is a difference, for notifying a client application of this difference.
2. Receiver according to Claim 1, characterized in that the demultiplexing and filtering means (5) are programmed following requests from the client application(s).
3. Receiver according to Claim 1 or 2, characterized in that the client application(s) comprise an electronic programme guide type application.
4. Receiver according to Claim 2, characterized in that an application determines a priority level for each request, the resources of the demultiplexing and filtering means (5) and means of storage (6) being reserved in the first instance for the requests having the highest priority level.
5. Receiver according to Claim 4, characterized in that there is implemented an advance priority level and a non-advance priority level, the non-advance priority level being higher than the advance priority level.
6. Receiver according to Claim 5, characterized in that the non-advance priority level is allocated to requests concerning data whose use by an application is certain, whereas the advance priority level is allocated to requests concerning data whose use by an application is probable, but not certain.
7. Receiver according to one of Claims 5 or 6, characterized in that the non-advance priority level is allocated to requests concerning data which are to be displayed as fast as possible.
8. Receiver according to one of Claims 2 to 7, characterized in that an application determines whether a request is of the permanent or one-off type, a request of the permanent type being maintained at the level of the programming of the demultiplexing and filtering means (5) until a contrary instruction from the application from which the permanent request originates, whereas a request of the one-off type is erased at the level of the programming of the demultiplexing and filtering means (5) after obtaining the corresponding data packet(s).
9. Receiver according to Claim 8, characterized in that the data stored in the storage means are data corresponding to permanent requests.

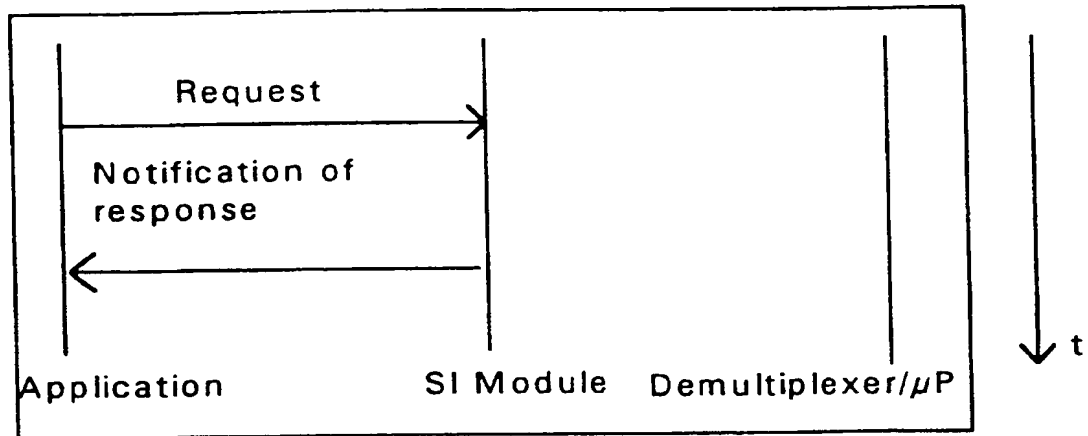


FIG. 2a

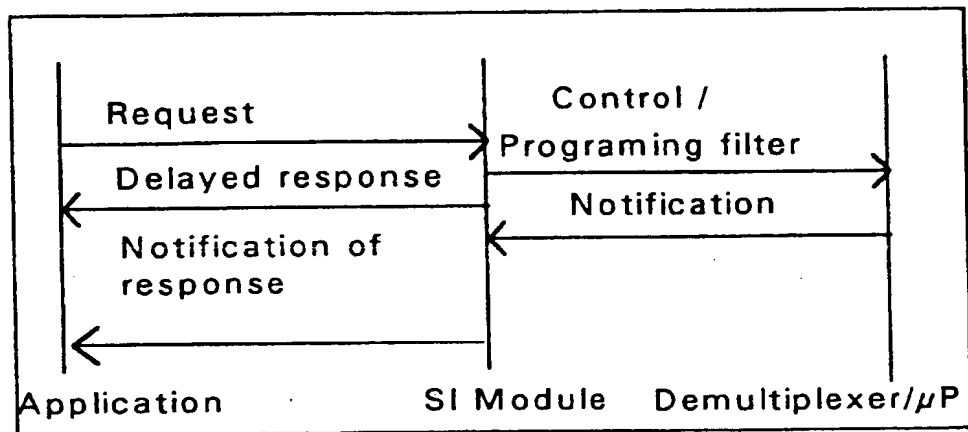


FIG. 2b

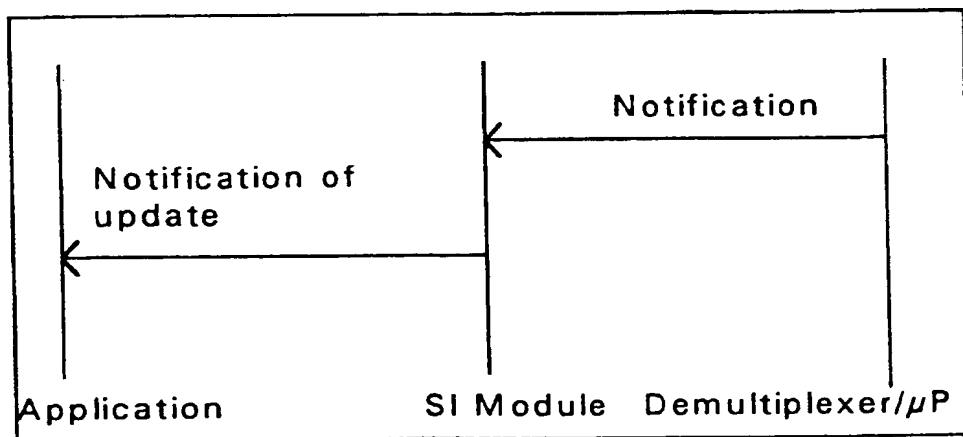


FIG. 2c

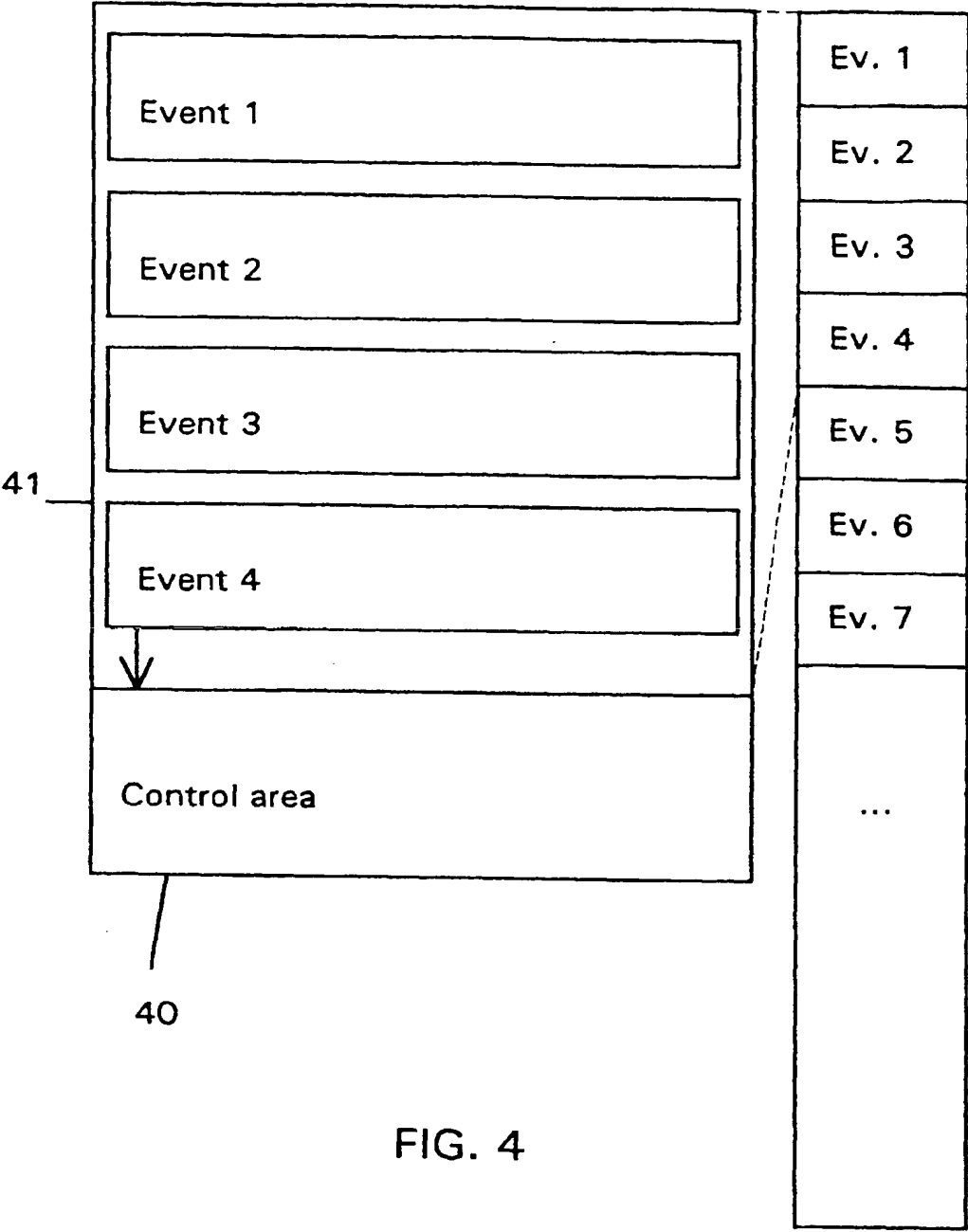


FIG. 4



European Patent
Office

EUROPEAN SEARCH REPORT

Application Number
EP 97 40 1903

| DOCUMENTS CONSIDERED TO BE RELEVANT | | | |
|---|--|-------------------|--|
| Category | Citation of document with indication, where appropriate, of relevant passages | Relevant to claim | CLASSIFICATION OF THE APPLICATION (Int.Cl.6) |
| Y | WO 93 09631 A (FINISAR CORPORATION) * the whole document * | 1-3 | H04H1/00 H04L12/18 H04N7/173 H04N7/52 H04N7/58 //H04N7/20 |
| Y | EP 0 679 028 A (THOMSON CONSUMER ELECTRONICS INC.) * the whole document * | 1-3 | |
| A | US 5 471 474 A (GROBICKI C. ET AL) * column 13, line 8 - line 21 * | 1-3 | |
| A | WO 95 31069 A (STARSIGHT TELECAST INC.) * page 8, line 24 - page 9, line 20 * * page 19, line 31 - page 20, line 20 * * page 164, line 11 - line 15 * | 1-3 | |
| A | WO 91 00670 A (THE SUPERGUIDE CORPORATION) * page 11, line 15 - page 14, line 20 * | 1-3 | |
| A | WO 94 14284 A (DISCOVERY COMMUNICATIONS INC.) * page 39, line 30 - page 79, line 14 * | 1-3 | |
| A | WO 96 04753 A (INTERIM DESIGN INC.) * page 39, line 12 - line 16 * | 1-3 | |
| A | US 5 510 844 A (CASH G. ET AL) * the whole document * | 1-3 | |
| The present search report has been drawn up for all claims | | | TECHNICAL FIELDS SEARCHED (Int.Cl.6) |
| Place of search THE HAGUE | | | Date of completion of the search 20 October 1997 |
| Examiner Verscheiden, J | | | |
| <p>CATEGORY OF CITED DOCUMENTS</p> <p>X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document</p> <p>T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document</p> | | | |

EPC FORM 1503 (03/82) (P/C/C/1)